



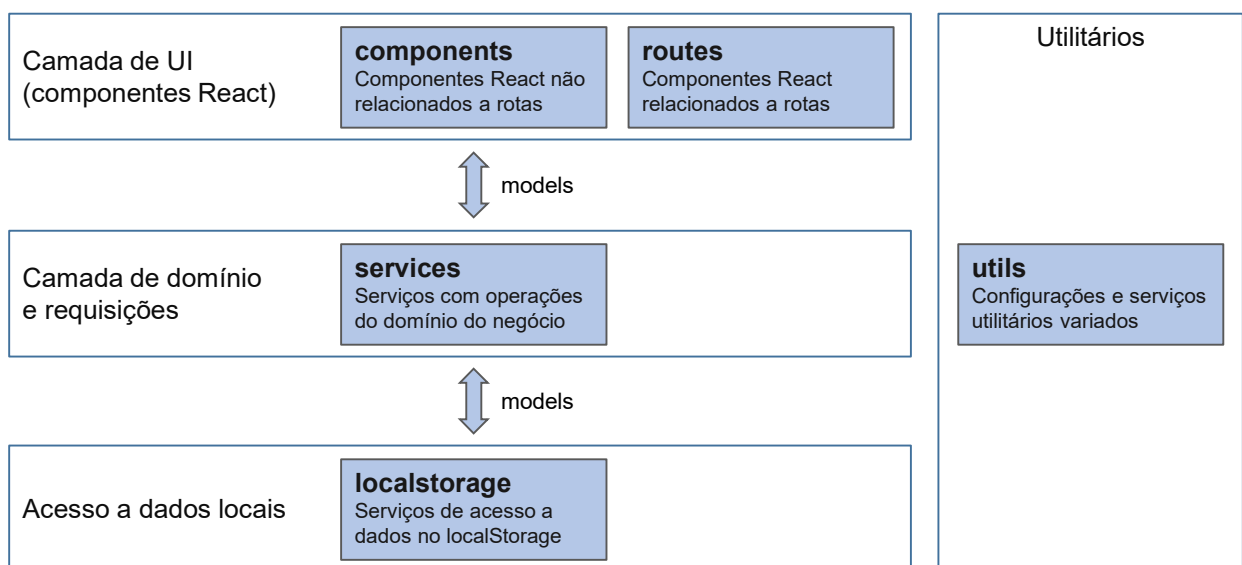
## Formação Desenvolvedor Moderno Módulo: Front end

Capítulo: Localstorage e lógica de carrinho

<https://devsuperior.com.br>

1

### Estrutura adotada para o projeto



2

2

# HTML Web Storage

Desde HTML5. Permite armazenar dados no navegador do usuário.

Diferentemente de cookies, o limite de armazenamento é bem maior, e a informação nunca é transferida para o servidor.

Possui escopo de ORIGEM (domínio e protocolo).

`window.localStorage`: persiste dados sem tempo de expiração

`window.sessionStorage`: dados são perdidos quando a aba é fechada

3

3

# JSON.stringify e JSON.parse

`JSON.stringify`: objeto => string

`JSON.parse`: string => objeto

4

4

# Projeto

Design Figma

<https://www.figma.com/file/ZrGNVNG0kZL6txDv4G8P6s/DSCommerce>

Documento de requisitos

[https://drive.google.com/drive/folders/1WTBggtq38cLeeQosPHjuhjSLxa94Lmx\\_](https://drive.google.com/drive/folders/1WTBggtq38cLeeQosPHjuhjSLxa94Lmx_)

Projeto HTML/CSS no Github:

<https://github.com/devsuperior/dscommerce-html-css>

5

5

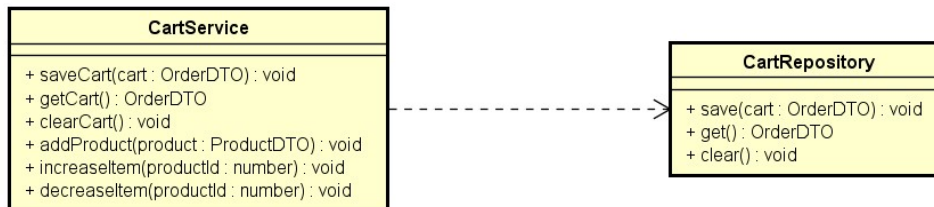
## Carrinho "mockado"

```
{
  items: [
    {
      productId: 4,
      quantity: 1,
      name: "PC Gamer",
      price: 1200,
      imgUrl: "https://raw.githubusercontent.com/devsuperior/dscatalog-resources/master/backend/img/4-big.jpg"
    },
    {
      productId: 5,
      quantity: 2,
      name: "Rails for Dummies",
      price: 100.99,
      imgUrl: "https://raw.githubusercontent.com/devsuperior/dscatalog-resources/master/backend/img/5-big.jpg"
    }
  ]
}
```

6

6

## Projeto dos componentes\*



\* não são componentes React, mas ainda sim são componentes do sistema, que na realidade serão implementados por modules do JS/TS

7

7

## Classes para models com dados calculados

```
export class OrderDTO {
  id?: number;
  items: OrderItemDTO[] = [];

  get total(): number {
    let sum = 0;
    this.items.forEach(item => {
      sum += item.subTotal;
    })
    return sum;
  }
}

export class OrderItemDTO {
  constructor(
    public productId: number,
    public quantity: number,
    public name: string,
    public price: number,
    public imgUrl: string
  ) {}

  get subTotal(): number {
    return this.price * this.quantity;
  }
}
```

8

8